AF/2155

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of: § Group Art Unit: 2155

Whay S. Lee § Examiner: Won, Michael Young

§ Atty. Dkt. No.: 5181-68300 P5974

Application No. 09/755,479 §

Filed: January 4, 2001 §

For: Scalable Routing Scheme for a Multi-Path Interconnection Fabric §

OIPE IAP79 FEB 0 2 2006 PATENT & TRADEMARK OFFICE

CERTIFICATE OF MAILING
37 C.F.R. § 1.8

I hereby certify that this correspondence is being deposited with the U.S. Postal Service with sufficient postage as First Class Mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date indicated below:

Robert C. Kowert
Name of Registered Representative

January 30, 2006 — Date

Signature

APPEAL BRIEF

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir/Madam:

Further to the Notice of Appeal filed November 28, 2005, Appellant presents this Appeal Brief. Appellant respectfully requests that the Board of Patent Appeals and Interferences consider this appeal.

02/03/2006 TBESHAH1 00000022 501505 09755479
01 FC:1402 500.00 DA

I. REAL PARTY IN INTEREST

The subject application is owned by Sun Microsystems, Inc., a corporation organized and existing under and by virtue of the laws of the State of Delaware, and having its principal place of business at 4150 Network Circle, Santa Clara, CA 95054.

II. RELATED APPEALS AND INTERFERENCES

No other appeals, interferences or judicial proceedings are known which would be related to, directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

III. STATUS OF CLAIMS

Claims 1-6, 9-22, 24-29, 32-42, 44-63, 65, 66, 68, 70 and 71 stand finally rejected. The rejection of claims 1-6, 9-22, 24-29, 32-42, 44-63, 65, 66, 68, 70 and 71 is being appealed. A copy of claims 1-6, 9-22, 24-29, 32-42, 44-63, 65, 66, 68, 70 and 71 is included in the Claims Appendix herein below.

IV. STATUS OF AMENDMENTS

No amendments to the claims have been submitted subsequent to the final rejection.

V. SUMMARY OF CLAIMED SUBJECT MATTER

Independent claim 1 is directed toward a method of sending messages in an interconnection fabric that couples together a plurality of nodes, each of which includes input ports and output ports. Each node in the fabric may have one or more ports connecting it with neighboring nodes. Input ports allow a node to receive messages from another node while output ports allow the node to send messages to other nodes. The method of claim 1 includes, for each of plurality of messages, dynamically selecting a route in the interconnection fabric from among a plurality of independent routes for

sending the message from a sending node to a destination node. For example, either the sending node in the fabric or a sending device connected to the sending node, according to different embodiments, may maintain a routing table containing different routing directive to use when sending message to various destination nodes. Each node in the interconnection fabric may be connected to each other node by multiple communication paths that form the fabric such that each communication path may be completely independent of each other path. In some embodiments, the interconnection fabric may include multiple, independent, paths and thus, the routing table may also contain multiple routing directive entries describing independent paths to each destination node. Thus, each node may have multiple paths to use when communicating with another node. Using multiple, independent, paths may allow a source node and a destination node to continue communicating even if one of the communication paths between the two nodes becomes inoperative. Dynamically selecting a route, as recited in claim 1, includes identifying a routing directive for the selected route from the sending node to the destination node. Part of dynamically selecting a route includes selecting different ones of the independent routes from the sending node to the destination node for at least two of the messages. When sending a message the node or the device may select an appropriate routing directive from the routing table. In some embodiments, a source node may select a route randomly from among the multiple paths and may try a different route if the destination node is unreachable using the first selected route. S*ee, e.g.*, FIGs. 1 and 18; page 5, lines 3-16; page 11, lines 12-26; page 13, lines 21-30; page 18, lines 4-13; page 24, lines 2-23; and page 26, lines 9-22.

The method also includes, for each message, encoding the routing directive in the message, wherein the routing directive describes the route and includes at least one segment, where each segment includes a direction component and a distance component. Either the sending node or the sending device connected to the sending node may encode the routing directive in the message. For example, the routing directive might be encoded as part of the header information contained in a message. A routing directive describes the route a message should take between a sending node and a destination node and may include a variable number of segments. A routing directive may include an indication of

the final segment with a special value of the direction component or the end of a routing directive may be indicated by the lack of any additional segment. The distance component and the direction component tell each node along the route how is should send the message. A distance component describes a distance in the interconnection fabric and a direction component specifies a direction in the interconnection fabric. For example, a distance component might be encoded as a binary number or as a number of tick marks. Different distance and direction components may be used with different interconnection fabric arrangement and topologies. S*ee, e.g.*, FIGs. 2, 4A, 4B, 5 and 8-13; page 5, lines 3-16; page 6, line 10 – page 7, line 2; page 11, lines 12-26; page 12, lines 12-22; page 24, lines 13-22; page 25, lines 2-18; page 26, lines 9-22; page 28, lines 2-29; page 29, lines 1-20; and page 30, lines 5-21.

The method also includes sending the message on one of the output ports of the sending node, receiving the message on one of the input ports of the first node connected to the output port of the sending node, decrementing the distance component for a current segment of the routing directive, selecting one of the output ports of the first node according to the current segment of the routing directive in the message, and sending the message on the selected one of the output ports of the first node. A message is sent on one of the sending node's output ports to one of the input ports of another node. For example, when a node receives a message, it looks to the routing directive for instructions on how to send the message. If the current segment is not yet completely executed, the node passes the message to the next node according to the current segment' instructions. If the current segment is complete, the node uses the next segment of the routing directive for instructions. For example, the output port of the sending node may be selected by checking to see if the distance component of the current segment is greater than zero. If so, an output port corresponding to the direction the message was traveling in when received is selected. Otherwise a different output port may be selected. In some embodiments, a direction component might expressly identify which port the node should use when sending the message. In other embodiments, the direction component might specify a direction relative to that in which the message was traveling when received. The distance directive is decremented to reflect that one hop along the route has been

made. The node may decrement the distance component of the current segment to reflect how much of the routing directive has been executed. *See, e.g.*, FIGs. 1, 6, 7 and 14-17; page 5, lines 3-29; page 6, lines 10-19; page 12, line 28 – page 13, line 23; and page 28, lines 2-29.

Independent claim 22 is directed to a node including a routing unit, a plurality of input ports and a plurality of output ports. The node is configured to be connected to an interconnection fabric that is configured to connect the node to a plurality of nodes. The routing unit is configured to receive a message being sent along a route from a sending node to a destination node in the interconnection fabric and is also configured to receive a routing directive encoded in the message. The routing directive describes the route and comprises at least one segment that includes a direction component and a distance component. The node is further configured to decrement the distance component of a current segment of the routing directive and to select one of the output ports according to the current segment. Please refer to the discussion of independent claim 1 above for a more detailed discussion regarding sending messages along a route in an interconnection fabric and regarding routing directives encoded in messages.

When the node of claim 22 is the sending node it is configured to dynamically select a route from among a plurality of independent routes from the sending node to the destination node. For at least two messages, the node is configured to dynamically select different ones of the independent routes from the sending node to the destination node when the node is the sending node. The node is also configured to encode the routing directive for the dynamically selected route in a message and to send the message on one of the output ports. Please refer to the discussion of independent claim 1 above for a more detailed discussion regarding selecting a route from among a plurality of independent routes and regarding sending messages including routing directives on an output port.

Independent claim 39 is directed toward a device including an interface configured to communicate with a source node in an interconnection fabric that includes

a plurality of routes between the source node and a destination node. The device of claim 39 also includes a controller configured to provide a first routing directive describing a first route from the source node to the destination node. The routing directive includes at least one segment that includes a distance component and a direction component. The distance component is configured to be decremented by a receiving node. The controller is also configured to encode the first routing directive in a message and to communicate the message to the source node to be sent on the interconnection fabric to the destination node. The controller is configured to maintain a routing table including a plurality of independent routes from the source node to the destination node and is further configured to dynamically select the first routing directive from the routing table when communicating the message to the source node to be sent on the interconnection fabric to the destination node. As the device of claim 39 is configured to perform actions similar to those described above regarding independent claim 1, please refer the discussion of claim 1 above for a more detailed discussion.

Independent claim 53 is directed toward a method of sending a message in an interconnection fabric that couples together a plurality of nodes, each of which includes a plurality of input ports and a plurality of output ports. The method of claim 53 includes identifying a route in the interconnection fabric for sending the message from a sending node to a destination node and encoding a routing directive in the message that describes the route and includes at least one segment. Each segment of the routing directive includes a direction component and a distance component. As described above regarding claim 1, each node in the interconnection fabric may be connected to each other node by multiple communication paths that form the fabric such that each communication path may be completely independent of each other path. Either a sending node or a sending device connected to the sending node may maintain a routing table containing different routing directive to use when sending message to various destination nodes. When sending a message the node or the device may select an appropriate routing directive from the routing table and then encode it in the message. The routing directive might be encoded as part of the header information contained in a message. A routing directive describes the route a message should take between a sending node and a destination node

and may include a variable number of segments. A routing directive may include an indication of the final segment with a special value of the direction component or the end of a routing directive may be indicated by the lack of any additional segment. Different distance and direction components may be used with different interconnection fabric arrangement and topologies. *See, e.g.*, FIGs. 2, 4A, 4B, 5 and 8-13; page 5, lines 3-16; page 6, line 10 – page 7, line 2; page 11, lines 12-26; page 12, lines 12-22; page 24, lines 13-22; page 25, lines 2-18; page 26, lines 9-22; page 28, lines 2-29; page 29, lines 1-20; and page 30, lines 5-21.

The method also includes identifying a return route from the destination node to the sending node and encoding a return routing directive in the message. A return routing directive describes a route from the destination node to the source node. The return routing directive may be identified along with the routing directive and may be included in a routing table. The sending node or the destination node may maintain a routing table describing a return route to the sending node. The message includes both the routing directive and the return routing directive when sent from the initial sending node. A return routing directive may be calculated by reversing the routing directive being used to send the message. The return route may be calculated incrementally. Each node might add information to the return routing directive as the message passes through that node. The sending node might encode an indication of the end of the route and each subsequent node might increment the distance component and may add a return direction component equal to the direction opposite the one used to send the message. As with the routing directives described above, a return routing directive includes at least one segment and each segment of the return routing directive includes a direction component and a distance component. Thus, a return route may be supplied by the source node, by the destination node, calculated on-the-fly, or by other means. *See, e.g.*, FIG. 14; page 6, lines 1-8; page 7, lines 4-14; and page 27, lines 7-30.

The method also includes receiving the message on one of the input ports of a first node connected to the output port of the sending node, decrementing the distance component for a current segment of the routing directive, selecting one of the output

ports of the first node according to the current segment of the routing directive in the message and sending the message on the selected output port of the first node. Please refer to the discussion above regarding claim 1 for a more detailed discussion regarding sending and receiving messages on output and input ports and decrementing the distance component for a current segment of the routing directive.

Independent claim 56 is directed to a node including a routing unit, a plurality of input port and a plurality of output ports. The node is configured to be connected to an interconnection fabric configured to connect the node to a plurality of nodes. Independent claim 59 is directed toward a device including an interface configured to communicate with a source node in an interconnection fabric that includes a plurality of routes between the source node and the destination node. As the node of claim 56 and the device of claim 59 are configured to perform actions similar to those described above regarding independent claim 53, please refer the discussion of claim 53 above for a more detailed discussion.

Independent claim 63 is directed to a method of sending a message in an interconnection fabric that couples together a plurality of nodes, each of which includes a plurality of input ports and a plurality of output ports. As described above regarding claim 1 and 53, the method of claim 63 includes identifying a route in the interconnection fabric for sending the message from a sending node to a destination node and encoding a routing directive in the message. The routing directive describes the route and comprises at least one segment, where each segment includes a direction component and a distance component. The method also includes sending the message on one of the output ports of the sending node and receiving the message on one of the input ports of the first node connected to the output port of the sending node. The method further includes decrementing the distance component for a current segment of the routing directive, selecting one of the output ports of the first node according to the current segment of the routing directive in the message, and sending the message on the selected one of the output ports of the first node. Please refer to the discussions of claim 1 and 53 above for a more detailed discussion regarding identifying and encoding routing directives as well

as regarding sending messages including routing directives.

The method also includes incrementally encoding a return routing directive in the message, including incrementing the distance component for a current segment of the return routing directive. The return routing directive describes a return route from the destination node to the sending node and includes at least one segment. Each segment of a routing directive includes a direction component and a distance component. For more details regarding incrementally encoding return routing directives, please refer to the discussion of claim 53 above.

If, after decrementing the distance component for a current segment of the routing directive, the distance component for the current segment of the routing directive is zero, the method further includes modifying the direction component of the current segment of the return routing directive and adding a new segment to the return routing directive so that the new segment becomes the current segment of the return routing directive when the message is sent on the selected output port. As described above, when routing a received message, a node may decrement the distance component of the current segment to reflect how much of the routing directive has been executed. When incrementally encoding a return routing directive, if after decrementing the distance component, if the distance component is zero, meaning that that portion of the routing directive has been completed, the node might add a return direction component equal to the direction opposite then one the sending route just completed. The node might create a new current return routing directive segment as well if the routing directive was not complete. In other words, if the message has not reached its destination node, the current node processing the message may add a new segment to the return routing directive, to which additional nodes may then add details, such as by incrementing the distance component of the new segment. *See, e.g.*, FIG. 14; page 6, lines 1-8; page 7, lines 4-14; and page 26, line 9 – page 27, line 30.

Independent claim 66 is directed to a node including a routing unit, a plurality of input ports and a plurality of output ports. The node is configured to be connected to an

interconnection fabric that is configured to connect the node to a plurality of nodes and also configured to perform actions similar to those described above regarding method claim 63. Please refer to the discussion above regarding claim 63 or a more detailed discussion.

Independent claim 68 is directed to a device including an interface configured to communicate with a source node in an interconnection fabric that includes a plurality of routes between the source node and a destination node. The device also includes a controller configured to provide a first routing directive describing a first route from the source node to the destination node. The routing directive includes at least one segment and each segment includes a distance component and direction component. The distance component is configured to be decremented by a receiving node. The controller is also configured to encode the first routing directive in a message and to communicate the message to the source node to be sent on the interconnection fabric to the destination node. The controller is further configured to incrementally encode a return routing directive describing a return route from the destination node to the source node in the message. The return routing directive describes a return route from the destination node to the sending node and includes at least one segment. Each segment of the return routing directive includes a direction component and a distance component. Please refer to the discussions above for a more detailed discussion regarding the above features of claim 68.

The return routing directive of claim 68 is also configured to be used to return an error message to the source node if a routing error is encountered. If a fault is encountered along the sending route, the partial return routing directive may provide a routing directive for sending an error message to the sending node. For instance, if a message fails to be sent to the destination node, the last receiving node may use the incrementally created return routing directive to return an error message to the sender. *See, e.g.*, page 7, lines 26-30; and page 27, lines 22-30.

Independent claim 71 is directed to a storage system includes a plurality of nodes

interconnected by an interconnection fabric where different ones of the nodes perform different functions in the storage system. Each one of a first portion of the nodes is a storage node includes at least one mass storage device and each one of second portion of the nodes is a host interface node configured to provide an interface for the storage system to a host computer. For example, FIG. 2 illustrates one embodiment of an interconnection fabric include multiple nodes, each of which may support different times of devices in a storage system. For example, some nodes may each be configured to support a controller such as a RAID controller. Other nodes may be configured with a host interface or a line card that may service as an interface to a host controller. Yet other nodes may be configured as routing nodes, mass storage nodes, or as storage cache memory nodes. *See, e.g.*, FIGs. 2, 3 and 4B; page 11, lines 3-10; page 12, lines 1-19; page 14, line 24 – page 15, line 6; and page 16, line 27 – page 17, line 20.

Each node includes a routing unit, a plurality input ports and a plurality of output ports. The routing unit of each node is configured to receive a message being sent along a route from a sending node to a destination node in the interconnection fabric. The routing unit of each node is also configured to receive a routing directive encoded in the message. The routing directive describes the route and includes at least one segment that includes a direction component and distance component. Each node is configured to receive the message on one of the input ports when the node is not the sending node and is further configured to decrement the distance component of a current segment of the routing directive and to select one of the output ports according to the current segment. Please refer to the discussions above for a more detailed discussion regarding routing directives, encoding routing directives in messages, and routing messages from a sending node to a destination node in an interconnection fabric.

The summary above describes various examples and embodiments of the claimed subject matter; however, the claims are not necessarily limited to any of these examples and embodiments. The claims should be interpreted based on the wording of the respective claims.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Claims 1, 9, 16, 18-25, 32, 37-39 and 41-46 stand finally rejected under 35 U.S.C. § 102(b) as being anticipated by Annapareddy et al. (U.S. Patent 5,602,839) (hereinafter "Annapareddy").

2. Claims 2-6, 17 and 26-29 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Annapareddy in view of Nugent (U.S. Patent 5,175,733).

3. Claims 10-13, 33-35 and 47-52 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Annapareddy in view of Walker et al. (U.S. Patent 5,613,069) (hereinafter "Walker").

4. Claims 36 and 40 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Annapareddy in view of Otterness et al. (U.S. Patent 6,792,472) (hereinafter "Otterness").

5. Claims 53-62, 68 and 70 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Flaig et al (U.S. Patent (5,105,424) (hereinafter "Flaig") in view of Walker.

6. Claims 14 and 15 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Annapareddy et al. (US 5,602, 839 A) in view of Walker et al. (US 5,613,069 A) and Nugent (US 5,175,733 A).

7. Claims 63, 65 and 66 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Flaig in view of Walker and Nugent.

8. Claim 71 stands finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Flaig in view of Brantley, Jr. et al. (U.S. Patent 4,980,822) (hereinafter "Brantley").

VII. ARGUMENT

First Ground of Rejection:

Claims 1, 9, 16, 18-25, 32, 37-39 and 41-46 stand finally rejected under 35 U.S.C. § 102(b) as being anticipated by Annapareddy et al. (U.S. Patent 5,602,839) (hereinafter "Annapareddy"). Appellant respectfully traverses this rejection in light of the following remarks.

Claims 1, 9, 16, 20, 22, 25, 32, 39, 41 and 44-46:

Regarding claim 1, Annapareddy fails to disclose encoding the routing directive in the message, wherein the routing directive describes the route and comprises at least one segment, wherein *each segment comprises a direction component and a distance component*. The Examiner's cites Fig. 4, Fig. 10, and column 6, lines 9-27 of Annapareddy. However, none of the Examiner's cited passages mention a routing directive segment including a direction component and a distance component. Figure 4 and column 6, lines 9-27 describe a message header that contains group and local node addresses and a deflection counter. None of the components of Annapareddy's message header correspond to a *direction* for a routing segment or to a *distance* for a routing segment. The address fields refer explicitly to source and destination node *addresses*, not to a distance or direction to take on the route from the source node to the destination node. Also, the deflection counter does not specify the distance to take in a specified direction. Instead, it decrements whenever a most-preferred path is not taken. If this counter decrements to zero before a message is delivered, the message delivery is terminated, thus preventing messages from endless traveling in the network without getting delivered (column 4, lines 14-30).

Figure 10, also cited by the Examiner, depicts a routing table containing preferred routes for sending messages between gateway nodes. Each entry specifies an absolute address for a source node and an absolute address for a destination node (column 9, lines

24-62), not a direction in which to travel from a source node toward a destination node. Therefore, the entries do not correspond to routing directive segments comprising *a direction component and a distance component*, as the Examiner contends. In fact, Annapareddy teaches away from a routing method using direction and distance components at column 2, lines 10-15 and 24-26, describing drawbacks to techniques that route messages in X and Y directions based on the difference between source and destination addresses in a mesh network. Annapareddy notes that such routing techniques are not transferable to other network topologies. **Annapareddy specifically notes that his system is in contrast to routing techniques that route messages in X and Y directions (column 2, lines 24-26).** Nowhere does Annapareddy describe a routing directive segment comprising a direction component and a distance component.

In the Advisory Action, the Examiner states that certain phrases used in Appellant's argument are not recited in the claims. The Examiner has misunderstood Appellant's argument. As noted above, Appellant's argument is that Annapareddy does not teach encoding a routing directive in the message, wherein the routing directive describes the route and includes at least one segment, wherein *each segment includes a direction component and a distance component*. The phrases referred to by the Examiner are intended to show that the Examiner's interpretation of Annapareddy is incorrect.

The Examiner also disagrees with the Appellant regarding Annapareddy's teaching away from using a routing directive including direction and distance components. The Examiner contends that Annapareddy is merely pointing out shortfalls of traditional routing system. The Examiner also asserts, "[n]owhere in column 2, lines 6-30, does Annapareddy explicitly recites [sic] that his invention does not employ **any** teachings of the prior art such as employing routing messages in X and Y directions" (bolding by Examiner). Appellant disagrees with the Examiner's interpretation of Annapareddy. As noted above, the background section of Annapareddy clearly outlines several drawbacks to routing messages using X and Y directives. For instance, Annapareddy states, "[u]nfortunately, these routing techniques [address-delta techniques] rely directly on network topology and therefore are not transferable from one network

topology to another or vice versa." Annapareddy concludes the background section by stating, "[t]herefore, there is a dire need for a message delivery and routing method and apparatus that is independent of a networking topology …". Thus, Annapareddy clearly states that address-delta techniques that "route messages in X and then Y directions" "rely directly on network topology" and further states that there is a "dire need" for a message delivery and routing method that is independent of a network topology. Annapareddy is clearly teaching away from routing messages using X and Y routing directives and is not merely teaching an objective of overcoming the shortfalls of prior art systems, as the Examiner contends.

In the Advisory Action, the Examiner again cites column 6, lines 12-21 as teaching a destination address field 210 that includes a group address field of which the group address 212 is used to route a message from a node of a source group to a destination group. The Examiner asserts, "[c]learly an address (group address 212) employed to route messages from a source node to a destination node teaches the limitation of a direction component." Appellant strongly disagrees with the Examiner's interpretation of Annapareddy. As noted above, the passage cited by the Examiner does not mention anything amount encoding a routing directive in a message that includes at least one segment that includes a direction component and a distance component. Furthermore, the cited passages states, and the Examiner agrees, that group addresses are used to route a message from a node of a source group to a destination group. However, Appellant disagrees with the Examiner's contention that a group address teaches a direction component. Annapareddy's group addresses cannot be considered either a direction or a distance component. Annapareddy teaches that when a node is routing a message, if the destination group address is not the same as the group address of the routing node, the node "selects message delivery directions from level-1 table 130 using the content of the group address 212 (block 550) as an index to select an entry in level-1 table 130" (parenthesis in original, Annapareddy, column 7, lines 56-61). There is no teaching in Annapareddy that group addresses correspond to directions. Annapareddy's group addresses are used as group identifiers and as indexes into a routing table *to obtain routing directions*. Annapareddy's group addresses are therefore not direction

components or distance components.

Additionally, Appellant's claim requires that a direction component and a distance component be included in a segment of a routing directive that is encoded in the message. Such is clearly not the case in Annapareddy's system. For example, FIG. 4 of Annapareddy clearly illustrates that the message header used to route messages includes a destination group address, a local node address, a deflection counter (which as shown above is not a direction component or a distance component), and a source address. Neither FIG. 4, nor any other portion of Annapareddy teaches anything about encoding in a message a routing directive that includes at least one segment that includes a direction component and a distance component.

The Examiner also cites column 3, lines 20-22 and lines 27-30, contending that Annapareddy teaches that the entries in a routing table are "ordered by the relative distance between the local node identified by that entry and node n". However, how Annapareddy orders the entries of his routing table is completely irrelevant to the fact that Annapareddy's system does not encode in messages a routing directive including at least one segment that includes a direction component and a distance component.

Annapareddy also fails to disclose <u>decrementing the distance component</u> for a current segment of the routing directive. The Examiner cites the deflection counter of Fig. 4 as implementing this feature. However, as discussed above and as described in the Examiner's own citations (column 4, lines 14-20 and column 11, lines 45-50), the deflection counter is not keeping track of the distance of a preferred routing segment. In contrast, it keeps track of deflected segments, those following a path other than the most-preferred path, in order to prevent endless traveling on the network.

Moreover, the Examiner's interpretation of Annapareddy clearly fails to meet the requirements of a rejection based on anticipation (i.e. a rejection under 35 U.S.C. § 102). Anticipation requires the presence in a single prior art reference disclosure of <u>each and every limitation</u> of the claimed invention, <u>arranged as in the claim</u>. M.P.E.P 2131;

Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick Co., 221 USPQ 481, 485 (Fed. Cir. 1984). The <u>identical</u> invention must be shown <u>in as complete detail</u> as is contained in the claims. *Richardson v. Suzuki Motor Co.*, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). As discussed above, Annapareddy fails to disclose encoding the routing directive in the message, wherein the routing directive describes the route and comprises at least one segment, wherein *each segment comprises a direction component and a distance component* and also fails to disclose decrementing the distance component for a current segment of the routing directive. Therefore, Annapareddy cannot be said to anticipate claim 1.

For at least the reasons above, the rejection of claim 1 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks as those made above regarding claim 1 also apply to independent claims 22 and 39.

<u>Claim 18</u>:

Regarding claim 18, Annapareddy fails to disclose wherein each direction component comprises <u>a direction relative to a routing direction the message was traveling in when received</u>. The Examiner cites Figures 5, 6, 10 and 11 each of which illustrates Annapareddy's level-1 and/or level-2 routing tables. However, these figures depict routing tables containing preferred routes for sending messages between various gateway nodes and local nodes using absolute addresses for source and destinations nodes (see, column 9, lines 24-62). Furthermore, the Examiner has apparently failed to consider the fact that claim 18, which depends from claim 1, requires that each direction component included in a routing directive *encoded in a message*, include a direction relative to a routing direction the message was traveling in when received. Thus, whether or not Annapareddy's routing tables include direction components is irrelevant. The Examiner has failed to cite any portion of Annapareddy that discloses the limitation of Appellant's claim 18.

Additionally, Annapareddy does not use direction components in his routing

tables. The figures cited by the Examiner clearly illustrate the Annapareddy's routing tables include group and node *addresses*, not direction components.

Thus, for at least the reasons above, the rejection of claim 18 is not supported by the cited art and removal thereof is respectfully requested.

Claim 19:

Regarding claim 19, Annapareddy fails to disclose wherein each direction component includes an identifier of one of the output ports of one of the nodes. The Examiner admits that Annapareddy does not explicitly disclose direction components that each include an identifier of an output port, but asserts that it is inherent in Annapareddy's system. Appellants disagree. First of all, as shown above regarding claim 1, Annapareddy fails to use direction components in segments of a routing directive. Secondly, Annapareddy does not mention anything about including an identifier for an output port in a direction component of a routing directive. Annapareddy uses I/O channels to send communicate betweens the nodes of his network, but does not disclose anything about including an identifier for an I/O channel in each direction component of a routing directive.

The Examiner only states that "such limitations are inherent" without providing any additional evidence or explanation regarding how the inclusion of output port identifiers in direction components of routing directives is inherent in Annapareddy's system. However, "[t]o serve as an anticipation when the reference is silent about the asserted inherent characteristic, such gap in the reference may be filled with recourse to extrinsic evidence." Additionally, "[s]uch evidence must make clear that the missing descriptive matter is necessarily present in the thing described in the reference, and that it would be so recognized by persons of ordinary skill." M.P.E.P 2131.01.III; *Continental Can Co. USA v. Monsanto Co.*, 948 F.2d 1264, 1268, 20 USPQ2d 1746, 1749 (Fed. Cir. 1991). The Examiner has failed to provide any evidence showing that Annapareddy's system inherently included an identifier of an output port in each direction component.

Thus, for at least the reasons above, the rejection of claim 19 is not supported by the cited art and removal thereof is respectfully requested.

Claims 21, 38 and 42:

Regarding claim 21, Annapareddy fails to disclose a destination node configured to communicate with a storage device, wherein the storage device comprises a disk drive. The Examiner cites figures 3 and 9 of Annapareddy. However, the cited figures do not teach a node configured to communicate with a disk drive. Instead, the cited figures teach that Annapareddy's nodes include system memory, such as DRAM, which is clearly different from a disk drive. Nowhere does Annapareddy mention anything about a destination node configured to communicate with a disk drive.

In the Advisory Action, the Examiner contends, erroneously, "[c]learly a disk drive is semiconductor memory." The Examiner is incorrect. As any one of ordinary skill in the art knows, a disk drive and semiconductor memory are two very different types of media. Furthermore, without some explicit teaching by Annapareddy that one of his node is configured to communicate with a storage device that includes a disk drive, Annapareddy cannot be said to anticipate Appellant's claim 21.

Thus, the rejection of claim 21 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks apply to claims 38 and 42.

Claim 24:

In regards to claim 24, contrary to the Examiner's assertion, Annapareddy fails to disclose that a node is configured to communicate with a device on a device port, where the device is configured to select a route, encode a routing directive in the message and communicate the message to the node on the device port, when the node is the sending node. The Examiner cites column 5, line 60 – column 6, line 8. However, the Examiner's citation merely describes the structure of a typical local node 100 of network

50, which consists of a processor, a memory containing two levels of routing tables, and some I/O channels. Annapareddy does not mention, either at the Examiner's cited portion or elsewhere, a sending node communicating with a device on a device port, where the device is configured to *select a route, encode a routing directive in the message* and *communicate the message to the node* on the device port.

Moreover, Annapareddy teaches that the processor within each node selects the route for message delivery using local routing tables (see, FIG. 7; column 6, lines 28-40; and column 7, lines 43-65). Thus, not only does Annapareddy fail to teach a sending node communicating with a device that selects a route and encodes a routing directive in a message, Annapareddy actually teaches away from a sending node communicating with such a device to select a route and encode a routing directive in a message.

In the Advisory Action, the Examiner states, "[t]he reference location was cited to teach the communication between the device and the node." However, as noted above, the cited passage does not mention anything about any communication between a device and one of Annapareddy's nodes. The cited passage does not mention anything about a node configured to communicate with a device on a device port.

The Examiner, in the Advisory Action, also asserts that Annapareddy teaches the use of a message header, determining a route by comparing the message with a table located at each receiving node, and selecting a delivery route. The Examiner concludes that through a combination of all these teaches, "Annapareddy explicitly teaches the recited limitations of claim 24." However, as noted above, Annapareddy teaches that each node consults a local routing table to determine how to route a message. This is clearly different than a device selecting a route, encoding a routing directive in the message and communicating a message to the node on a device port of the node when the node is the sending node.

For at least the reasons above, the rejection of claim 24 is not supported by the cited art and removal thereof is respectfully requested.

<u>Claim 37</u>:

Regarding claim 37, Annapareddy fails to disclose a destination node configured to <u>communicate with a mass storage device</u>. The Examiner cites figures 3 and 9 of Annapareddy. However, the cited figures do not teach a node configured to communicate with a mass storage device. Instead, the cited figures teach that Annapareddy's nodes include system memory, such as DRAM, which is clearly different from a mass storage device as this term is understood in the art. Nowhere does Annapareddy mention anything about a destination node configured to communicate with a mass storage device. Without some specific teaching by Annapareddy that his nodes communicate with a mass storage device, Annapareddy cannot be said to anticipate claim 37. Thus, the rejection of claim 37 is not supported by the cited art and removal thereof is respectfully requested.

<u>Claim 42</u>:

Regarding claim 42, Annapareddy fails to disclose a controller that includes a disk storage device controller. The Examiner cites figures 3 and 9 of Annapareddy. However, the cited figures do not illustrate any controller including a disk storage device controller. Instead, the cited figures teach that Annapareddy's nodes include a process coupled to memory and to multiple I/O channels, but fail to mention anything regarding a controller including a disk storage device controller. In fact, nowhere does Annapareddy mention anything about a controller including a disk storage device controller. Thus, the rejection of claim 42 is not supported by the cited art and removal thereof is respectfully requested.

<u>Second Ground of Rejection</u>:

Claims 2-6, 17 and 26-29 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Annapareddy in view of Nugent (U.S. Patent 5,175,733). Appellant respectfully traverses this rejection for at least the reasons presented below.

Claims 2, 5 and 26:

Regarding claim 2, Annapareddy in view of Nugent fails to teach or suggest where selecting an output port comprises if, after decrementing the distance component for a current segment of the routing directive, the distance component for the current segment is greater than zero, selecting the output port corresponding to a same routing direction in which the message was traveling when received and if, after said decrementing, the distance component of the current segment is zero, selecting the output port corresponding to the direction component of the current segment. The Examiner admits that Annapareddy does not teach selecting the output port based upon the direction component and upon whether or not decrementing a distance component results in a distance component of zero or not. The Examiner relies upon Nugent, citing FIG. 8 and column 14, line 1 – column 15, line 14.

The Examiner contends that it would have been obvious to a person of ordinary skill in the art to employ the teachings of Nugent within the system of Annapareddy because decrementing the directional component to zero allows directional limits to be set thereby triggering a change in directions such as from X-direction to Y or Z-direction. However, Annapareddy does not use directional or distance components in his routing scheme. Instead, Annapareddy relies upon absolute node addresses for source and destination nodes (Annapareddy, column 2, lines 60 – 67; column 6, lines 9-27). It would not make sense to modify Annapareddy to include the distance and direction based routing system of Nugent. Furthermore, as noted above, Annapareddy teaches away from using direction and distance components in a routing directive. Thus, modifying Annapareddy to use the distance and directional components of Nugent would render the resultant system unsatisfactory for its intended purpose. As noted above, Annapareddy specifically teaches that his absolute address based routing scheme is intended to be independent of any network topology and further teaches that using distance and directional components to specify a route renders the route dependent upon the specific

network topology (Annapareddy, column 2, lines 6-31). Thus, the Examiner has failed to provide proper motivation to combine the teachings of Annapareddy and Nugent.

In the Advisory Action, the Examiner relies upon his (erroneous) contention in the rejection of claim 1 that Annapareddy teaches the use of a routing directive encoded in a message and including a segment that includes a direction component and a distance component to provide motivation to combine Annapareddy with Nugent. However, as shown above regarding claim 1, Annapareddy clearly fails to teach the use of direction and distance components. Thus, the Examiner has failed to provide a proper motivation to combine Annapareddy and Nugent.

Therefore, for at least the reasons above, the rejection of claim 2 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks apply to claims 5 and 26.

<u>Claims 3 and 27</u>:

Regarding claim 3, Annapareddy in view of Nugent fails to teach or suggest wherein if, after decrementing, the distance component for the current segment is zero, and the output port is selected according to the direction component of the current segment, <u>removing the current segment from the routing directive so that a next segment becomes the current segment when the message is sent on the selected output port</u>. The Examiner replies on Nugent claiming that removing the current segment from a routing directive so that a next segment becomes the current segment when the message is sent on the selected output port is inherent in Nugent's system.

The Examiner has also failed to provide any evidence demonstrating that Nugent's system necessarily includes removing a current segment from a routing directive so that a next segment becomes the current segment when the message is sent on the selected output port. The Examiner only states that Nugent's system inherently includes removing the current segment from a routing directive so that a next segment

becomes the current segment when the message is sent on the selected output port.

Additionally, Nugent teaches the use of a single segment routing directive in messages (Nugent, column 8, lines 40-45). For example, a routing directive may take the from: (X, Y) or (X,Y,Z). Nugent teaches decrementing each component (either X, Y or Z) at each node as the message is routed. Nugent's system does not include removing any segment of the routing directive, however. In fact, Nugent's system relies upon the fact that a node can first compare the X component to zero to determine whether or not the message needs to travel along an X direction. If the X component is zero, the message does not need to be routed in the X dimension and the message is check to see if it requires movement in the Y dimension. (Nugent, column 13, line 61 – column 14, line 18). Thus, Nugent's system does not inherently include removing a current segment of a routing directive, contrary to the Examiner's assertion.

Furthermore, as noted above regarding claim 2, there is no motivation to combine the teachings of Annapareddy and Nugent. In fact, since Annapareddy clearly teaches routing messages by using two-level routing tables located on each of Annapareddy's nodes, modifying Annapareddy to include the X and Y based directional routing of Nugent would necessarily run counter to the intended use of Annapareddy's system. Thus, modifying Annapareddy to use the distance and directional components of Nugent would render the resultant system unsatisfactory for its intended purpose. As such, the combination of Annapareddy and Nugent suggest by the Examiner is improper.

For at least the reasons above, the rejection of claim 3 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks apply to claim 27 as well.

Claims 4 and 28:

Regarding claim 4, Annapareddy in view of Nugent fails to teach or suggest a routing directive that includes <u>a pointer to the current segment</u>, and wherein removing the

current segment includes moving the pointer to the next segment. The Examiner has not cited any portion of the prior art that teaches or suggested the limitations of claim 4. Instead, the Examiner asserts that a routing directive including a pointer to the current segment and that moving the pointer to the next segment as a part of removing the current segment are inherent in Annapareddy's system. Appellants strongly disagree with the Examiner's assertion. Firstly, the Examiner has filed to provide any evidence or interpretation as to why such features would be inherent in Annapareddy. Secondly, Annapareddy's system does not include segmented routing directives encoded in a message and also does not include a pointer to the current segment as part of a routing directive. As shown above regarding claim 1, Annapareddy teaches the use of routing tables located at each node to route messages. The messages in Annapareddy do not include a routing directive including segments. Instead, a message in Annapareddy only includes the destination node and group node addresses that are used as indexes into the routing tables in each node (Annapareddy, column 5, line 60-column 6, line 27).

Additionally, Annapareddy does not teach or suggest removing a current segment of the routing directive. In fact, Annapareddy does not teach or suggest removing any routing information from a message header during routing. Annapareddy includes the destination node and group address in messages and removing any portion of Annapareddy's routing information would necessarily prevent a message from being delivered.

Furthermore, the Examiner, regarding the rejection of claim 3, discussed above, relies (erroneously) upon Nugent to teach removing a current segment of the routing directive. Thus, is does not make sense for Annapareddy to inherently include updating a pointer to a next segment as part of removing the current segment if Annapareddy does not include removing the current segment. If Annapareddy's does not remove the current segment (which is doesn't, as Annapareddy's system does not use routing directive segments) Annapareddy's system would have no need to update a pointer from a current segment being removed to a next segment. Thus, the Examiner has clearly failed to provide a proper rejection of claim 4.

It is clearly not inherent in Annapareddy's system to include a routing directive that includes a pointer to a current segment of the routing directive. Nor is it inherent in Annapareddy's system to move the pointer to a next segment of the routing directive as part of removing the current segment. Nugent also fails to mention anything about updating a routing directive including a pointer to the current segment or about moving the pointer to the next segment when removing the current segment. Nugent using a single segment routing directive, and as such, has no need for a pointer to a current segment, for removing a current segment, or for moving the pointer to the next segment when removing the current segment. Thus, the Examiner's combination of Annapareddy and Nugent fails to teach or suggest a routing directive that includes <u>a pointer to the current segment</u>, and wherein removing the current segment includes <u>moving the pointer to the next segment</u>. The rejection of claim 4 is not supported by the cited art and removal thereof is respectfully requested.

<u>Claims 6, 17 and 29</u>:

Regarding claim 6, Annapareddy in view of Nugent fails to teach of suggest the subsequent node selecting the corresponding output port <u>if the direction component for the current segment specifies a routing direction</u>; and selecting a device port if the direction component for the current segment specifies that the subsequent node is the destination for the message, contrary to the Examiner's contention. The Examiner cites column 7, lines 53-56 and column 9, lines 1-23 of Annapareddy. However, these portions of Annapareddy fail to mention anything about selecting between an <u>output port</u> and a <u>device port</u> dependent upon the direction component of the current segment. Instead, they describe that routing is complete when a message destination node address equals the current local node address and that various I/O channels connect nodes in the network. The cited passages do not mention device ports at all. Nugent is not relied upon by the Examiner to teach anything about selecting an output port or a device port according to whether the direction component for the current routing directive segment specifies a routing direction or that the subsequent node is the destination node.

Therefore, neither Annapareddy nor Nugent, nor the combination of the two, teaches or suggests selecting between an output port and a device port dependent on the direction component of a routing segment.

In the Advisory Action, the Examiner asserts, "ports are inherent based on I/O channels taught." However, the Examiner apparently misunderstood Appellant's argument. Appellant's are not arguing that Annapareddy's nodes do not contain any ports. The Examiner is clearly oversimplifying the limitations of Appellant's claim. Claim 6 recites a subsequent node selecting an output port of the direction component for the current segment of the routing directive specifying a routing direction and selecting a device port if the direction component for the current segment specifies that the subsequent node (i.e. the same node performing the selection) is the destination for the message. In other words, when a node is selecting how to route a message it selects *an output port* if the selecting node is not the destination and <u>selects a device port</u> if the selecting node is the destination. Annapareddy fails to teach such a selection when routing a message.

Thus, for at least the reasons above, the rejection of claim 6 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks apply to claims 17 and 29.

<u>Third Ground of Rejection</u>:

Claims 10-13, 33-35 and 47-52 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Annapareddy in view of Walker et al. (U.S. Patent 5,613,069) (hereinafter "Walker"). Appellant respectfully traverses this rejection for at least the following reasons.

<u>Claims 10-12 and 33-35</u>:

Regarding claim 10, in contrast to the Examiner's assertion, Annapareddy in view

of Walker fails to teach or suggest encoding a return routing directive in the message, identifying a return route from the destination node to the sending node, where the return routing directive describes the return route and comprises at least one segment, and where each segment comprises a direction component and a distance component.

The Examiner contends that Annapareddy uses routing directive segments that include direction and distance components. However, as discussed above regarding claim 1, the message header of Annapareddy does not comprise distance and direction components, but instead uses absolute addresses of the sending and destination nodes.

The Examiner relies upon Walker to teach identifying a return route from the destination node to the sending node and encoding a return routing directive in the message. However, the return directive in Walker is based on independent routelets that define an absolute switching path that depends only on the hardware configuration of the node (col. 7, lines 24-31). Walker states that each routelet defines the absolute path at each node. The routelet-based mechanism of Walker specifically does not use direction and distance components. It would not make sense to apply the routelet-based return route encoding of Walker to the routing mechanism of Annapareddy.

Thus, since Annapareddy and Walker, either singly or in combination, fail to teach or suggest encoding a return routing directive comprising segments that include direction and distance components. For at least the reasons above, the rejection of claim 10 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks apply to claim 33.

Claim 13:

Annapareddy in view of Walker fails to teach or suggest incrementally encoding a return routing directive in the message, wherein the return routing directive describes a return route from the destination node to the sending node and comprises at least one segment, and wherein each segment comprises a direction component and a distance

component. The Examiner cites column 5, lines 20-25 of Walker that describes an automatically generated packet trailer that records the return route to the packet originator at each stage through Walker's network. However, as noted above regarding claim 10, Walker's routelet-based routing mechanism does not use direction and distance components. Thus, Walker fails to teach incrementally encoding a return routing directive that includes direction and distance components.

As described above regarding claim 1, Annapareddy also fails to use direction and distance components as part of a routing directive encoded in messages. Annapareddy also fails to mention anything about encoded a return routing directive in a message. Thus, Annapareddy fails to teach or suggest incrementally encoding a return routing directive in the message, wherein the return routing directive describes a return route from the destination node to the sending node and comprises at least one segment, and wherein each segment comprises a direction component and a distance component.

Furthermore, the Examiner's combination of Annapareddy and Walker fails to result in a system that teaches or suggests incrementally encoding a return routing directive that includes at least one segment, where each segment includes a direction component and a distance component. Since, as noted above, neither Annapareddy nor Walker teaches or suggest using direction and distance components included in segments of an incrementally encoded return routing directive, the combination of Annapareddy and Walker also fails to teach or suggest using direction and distance components included in segments of an incrementally encoded return routing directive. Instead, the Examiner's combination of Annapareddy an Walker results in a system that would build a return routing directive that either uses absolute address, either node and group addresses as taught by Annapareddy or the absolute switching addresses taught by Walker.

Thus, for at least the reasons above, the rejection of claim 13 is not supported by the cited art and removal thereof is respectfully requested.

Claims 47-50:

Claims 47-50 are allowable for at least the reasons presented above regarding their respective independent claims.

Claim 51:

Regarding claim 51, Annapareddy in view of Walker fails to teach or suggest wherein the return routing directive is configured to be used to return an error message to the source node if a routing error is encountered. The Examiner cites column 9, lines 57-62 of Walker. However, **Walker teaches away** from using the return routing directive to return an error message. Walker teaches that his system does not "handle error detection and correction" (Walker, column 5, lines 30 – 35). In fact, the Examiner's cited passage states that if a packet is misrouted or is corrupted preventing delivery, Walker's system "will discard the packet and *take no other action*." Thus, Walker clearly teaches away from using a return routing directive to return an error message to the source node if a routing error is encountered.

Annapareddy does not mention anything about error messages or about using a return routing directive to return an error message to the source node if a routing error is encountered. Thus, the combination of Annapareddy and Walker would clearly fail to teach or suggest a return routing directive configured to be used to return an error message to the source node if a routing error is encountered.

Furthermore, since Walker teaches away from using a return routing directive to return an error message, the Examiner has failed to provide a *prima facie* case of obviousness. "A *prima facie* case of obviousness can be rebutted if the applicant...can show that the art in any material respect 'taught away' from the claimed invention...A reference may be said to teach away when a person of ordinary skill, upon reading the reference...would be led in a direction divergent from the path that was taken by the applicant." *In re Haruna*, 249 F.3d 1327, 58USPQ2d 1517 (Fed. Cir. 2001). Additionally, a reference "may be said to teach away when a person of ordinary skill,

upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant." *In re Gurley*, 27 F.3d 551, 553 (Fed. Cir.1994) (*emphasis added*).

Thus, for at least the reasons above, the rejection of claim 51 is not supported by the cited art and removal thereof is respectfully requested.

Claim 52:

Regarding claim 52, Annapareddy in view of Walker fails to teach or suggest a controller configured to use the incrementally created return routing directive to locate the routing error if an error message is returned, wherein the incrementally created return routing directive indicates a last node that successfully received the message. The Examiner cites Walker, column 9, lines 57-62. However, as noted above regarding the rejection of claim 51, the cited portion of *Walker teaches away* from delivering error messages, whether using a return routing directive or any other technique. In fact, Walker clearly states that his system does not perform any error correction or detection (Walker, column 5, lines 30 – 35).

Annapareddy also fails to teach or suggest anything about using a return routing directive to locate a routing error or about a return routing directive that indicate a last node that successfully received the message. Thus, the combination of Annapareddy and Walker also fails to teach or suggest a controller configured to use an incrementally created routing directive to locate a routing error if an error message is returned. Additionally, since Walker teaches away from locating a routing error (column 5, lines 30 – 35), the Examiner has failed to provide a *prima facie* obviousness rejection.

Furthermore, even if the combination of Annapareddy and Walker taught locating a routing error using an incrementally created return routing directive, which it doesn't, the combination would not locate a routing error if an error message is return, as recited in Appellant's claim 52. Since, as illustrated above regarding the rejection of claim 51,

the combination of Annapareddy and Walker does not teach or suggest returning an error message, no combination of Annapareddy and Walker could include locating a routing error *if an error message is returned.*

Thus, for at least the reasons above, the rejection of claim 52 is not supported by the cited art and removal thereof is respectfully requested.

Fourth Ground of Rejection:

Claims 36 and 40 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Annapareddy in view of Otterness et al. (U.S. Patent 6,792,472) (hereinafter "Otterness"). Appellant respectfully traverses this rejection of claims 36 and 40 for at least the reasons presented above regarding their respective independent claims.

Fifth Ground of Rejection:

Claims 53-62, 68 and 70 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Flaig et al (U.S. Patent (5,105,424) (hereinafter "Flaig") in view of Walker. Appellant respectfully traverses this rejection for at least the reasons presented below.

Claims 53-59 and 61:

Regarding claim 53, Flaig in view of Walker teaches fails to teach or suggest identifying a return route from the destination node to the sending node and encoding a return routing directive in the message, wherein the message includes both the routing directive and the return directive when sent from the initial sending node, in contrast to the Examiner's assertion.

The Examiner admits that Flaig does not teach encoding a return routing directive in the message. The Examiner refers to Walker in regard to this feature of claim 53. However, Walker specifically states that the message is initially sent without a return

directive ("routing trailer is null"). In Walker, the return directive is dynamically generated only after the message has been sent from the initial sending node. *See*, Walker – col. 8, lines 15-24. Therefore, Walker specifically does not teach or suggest that the message includes both the routing directive and the return routing directive when sent from the initial sending node, as is recited in claim 53.

In the Advisory Action, the Examiner argues that Walker's describes that when a packet "enters the network, … the routing trailer is null" is not equivalent to "when sent from the initial sending node" as recited in the claim. The Examiner further cites column 7, lines 62-64 of Walker and refers to the fact that Walker's packet structure includes a routing trailer that describes where the packet has come from. However, the Examiner's own citations (above, and column 5, lines 20-25) clearly describe that a packet trailer is <u>dynamically generated</u> and <u>records the return route as the packet flows through the network</u>. Thus, it cannot be included in the message <u>when sent</u> from the initial sending node. Walker's statement that when a packet enters the network the routing trailer is null clearly indicates that Walker does not encode a routine routing directive describing a return route from the destination node to the sending node in the message. As noted above, Walker very clearly describes building the return route as the message travels through the network.

Furthermore, the Examiner's argument that Walker's statement that a routing trailer is null when a packet enters the network is not equivalent to "when sent from the initial sending node" is irrelevant to the fact that Walker's system builds the return routing directive as the message progresses through the network (Walker, column 5, lines 20-25, and column 8, lines 15-25). Thus, Walker fails to teach or suggest encoding a routine routing directive describing a return route from the destination node to the sending node in the message wherein the message includes both the routing directive and the return routing directive when sent from the initial sending node.

Therefore, Walker clearly fails to overcome the deficiencies of Flaig. Thus, the combination of Flaig in view of Walker also fails to teach or suggest encoding a routine

routing directive describing a return route from the destination node to the sending node in the message wherein the message includes both the routing directive and the return routing directive when sent from the initial sending node, as recited in claim 53.

For at least the reasons above, the rejection of claim 53 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks as those above regarding claim 53 also apply to claims 56 and 59.

Claim 60:

Regarding claim 60, Flaig in view of Walker fails to teach or suggest a controller configured to select the return routing directive from a routing table. The Examiner cites column 4, lines 46-62, but fails to mention whether this passages is cited from Flaig or Walker. However, neither Flaig nor Walker describes selecting a return routing directive from a routing table. Flaig does not mention anything at all about selecting a return routing directive, whether from a routing table or elsewhere. Walker also does not teach selecting a return routing directive from a routing table. Instead, Walker teaches incrementally building a return routing directive as a message is routed through the network (Walker, column 5, lines 20-25 and column 8, lines 15-25).

Additionally, the combination of Flaig in view of Walker does not teach or suggest selecting a return routing directive from a routing table. Instead, the combination of Flaig in view of Walker would clearly use Walker's incremental approach to building a return routing directive rather than selecting a return routing directive from a routing table, as suggested by the Examiner.

Thus, for at least the reasons above, the rejection of claim 60 is not supported by the cited art and removal thereof is respectfully requested.

<u>Claim 62</u>:

Regarding claim 62, Flaig in view of Walker fails to teach or suggest a return routing directive configured to be used to return an error message to the source node if a routing error is encountered. The Examiner cites Walker column 9, lines 57-62. However, as illustrated above regarding claim 51, **Walker teaches away** from using the return routing directive to return an error message. Walker teaches that his system does not "handle error detection and correction" (Walker, column 5, lines 30 – 35). In fact, the Examiner's cited passage states that if a packet is misrouted or is corrupted preventing delivery, Walker's system "will discard the packet and *take no other action*." Thus, Walker clearly teaches away from using a return routing directive to return an error message to the source node if a routing error is encountered.

Flaig fails to mention anything about a return routing directive configured to be used to return an error message to the source node if a routing error is encountered. If fact, Flaig fails to teach or suggest anything at all about error message. Thus, neither Flaig nor Walker, nor any combination of Flaig and Walker teach or suggest a return routing directive used to return an error message if a routing error is encountered.

Furthermore, since Walker teaches away from using a return routing directive to return an error message, the Examiner has failed to provide a *prima facie* case of obviousness. "A *prima facie* case of obviousness can be rebutted if the applicant...can show that the art in any material respect 'taught away' from the claimed invention...A reference may be said to teach away when a person of ordinary skill, upon reading the reference...would be led in a direction divergent from the path that was taken by the applicant." *In re Haruna*, 249 F.3d 1327, 58USPQ2d 1517 (Fed. Cir. 2001). Additionally, a reference "may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or <u>would be led in a direction divergent from the path that was taken by the applicant</u>." *In re Gurley*, 27 F.3d 551, 553 (Fed. Cir.1994) (*emphasis added*).

Thus, for at least the reasons above the rejection of claim 62 is not supported by the cited art and removal thereof is respectfully requested.

Claim 68:

In regard to claim 68, Flaig in view of Walker does not teach or suggest incrementally encoding a return routing directive describing a return route from the destination node to the source node in the message, wherein the return routing directive describes a return route from the destination node to the sending node and comprises at least one segment, and wherein each segment comprises a direction component and a distance component. The Examiner relies upon Walker to incrementally encode such a return routing directive. However, the return directive in Walker is based on independent routelets that define an absolute switching path (Walker – col. 7, lines 24-20. The routelet-based mechanism of Walker specifically does not use direction and distance components. As Flaig uses a completely different method of routing directives, it would not make sense to apply the routelet-based return route encoding of Walker to the routing mechanism of Flaig. Therefore, the combination of Flaig and Walker would not result in a system that included a return routing directive that uses distance and direction components. Instead, the combination of Flaig and Walker would use the absolute switching path routelets of Walker.

In the Advisory Action, the Examiner respond by pointing out that "one cannot show nonobviousness by attacking the references individually where the rejections are based on combinations of references." However, Appellants specifically argued (repeated above) that the *combination* of Flaig in view of Walker would not result in a system that included a return routing directive that uses distance and direction components.

Furthermore, Flaig in view of Walker does not teach or suggest a return routing directive configured to be used to return an error message to the source node if a routing error is encountered. The Examiner cites column 2, line 42-44 where Walker states that

detection of errors and re-transmission of data are mandatory in almost all computer applications. In the Response to Arguments section, the Examiner reiterates his assertion that this one statement in Walker suggests "returning an error message". However, a simple statement that applications detect errors and re-transmit data does not imply the specific limitation of incrementally encoding a return routing directive configured to be used to return an error message to the source node if a routing error is encountered.

Furthermore, the portion cited by the Examiner is from the background section of Walker's disclosure and does not refer to the rest of Walker's teachings. In fact, Walker specifically states that his system "does not handle error detection and correction" (column 5, lines 32-33). Thus, **Walker teaches away** from error detection and correction.

In the Advisory Action, the Examiner responds by arguing that "Walker is referring to the Data Link Layer of the OSI model in which error detection and correction is traditionally associated with" and by asserting that "although Walker does not explicitly teach the limitation above, the combinational teachings ... clearly suggest such limitation." Appellants disagree. Firstly, whether or not Walker refers to the Data Link Layer of the OSI model is irrelevant. Claim 68 is not rejected over the Data Link Layer of the OSI model. Secondly, the Examiner has provided any evidence that the Data Link Layer of the OSI model includes a incrementally encoded return routing directive, as recited in claim 68, and configured to be used to return an error message to the source node. Thirdly, the Examiner's rejection of claim 68 relies on Walker to teach incrementally generating a return routing directive and it would not make since for the Data Link Layer of the OSI model to use a specific structure from Walker's system when performing its own error detection and correction. Therefore, the Examiner's combination of Flaig and Walker fails to teach or suggest a return routing directive configured to be used to <u>return an error message to the source node if a routing error is encountered</u>.

Thus, the rejection of claim 68 is not supported by the cited art and removal

thereof is respectfully requested.

<u>Claim 70</u>:

Regarding claim 70, Flaig in view of Walker fails to teach or suggest <u>using the incrementally created return routing directive to locate the routing error</u> if an error message is returned, wherein the incrementally created return routing directive indicates a last node that successfully received the message. The Examiner cites column 9, lines 57-62 of Walker. However, as noted above, the cited portion of *Walker teaches away* from delivering error messages, whether using a return routing directive or any other technique. In fact, Walker clearly states that his system does not perform any error correction or detection (Walker, column 5, lines 30 – 35).

Flaig also fails to teach or suggest anything about using a return routing directive to locate a routing error or about a return routing directive that indicate a last node that successfully received the message. Thus, the combination of Flaig and Walker also fails to teach or suggest a controller configured to use an incrementally created routing directive to locate a routing error if an error message is returned. Additionally, since Walker teaches away from locating a routing error (column 5, lines 30-35), the Examiner has failed to provide a *prima facie* obviousness rejection.

Furthermore, even if the combination of Flaig and Walker taught locating a routing error using an incrementally created return routing directive, which it doesn't, the combination would not locate a routing error if an error message is return, as recited in Appellant's claim 70. Since, as illustrated above regarding the rejection of claim 62, the combination of Flaig and Walker does not teach or suggest returning an error message, no combination of Flaig and Walker could include locating a routing error *if an error message is returned*.

Thus, for at least the reasons above, the rejection of claim 70 is not supported by the cited art and removal thereof is respectfully requested.

Sixth Ground of Rejection:

Claims 14 and 15 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Annapareddy et al. (US 5,602,839) in view of Walker et al. (US 5,613,069) and Nugent (US 5,175,733). Appellant respectfully traverses this rejection for at least the following reasons.

Claim 14:

Regarding claim 14, Annapareddy in view of Walker and Nugent fails to teach or suggest wherein if, after decrementing, the distance component for the current segment of the routing directive is zero, modifying the direction component of a current segment of the *return* routing directive and adding a new segment to the *return* routing directive so that the new segment becomes the current segment of the return routing directive when the message is sent on the selected output port. The Examiner asserts that Nugent teaches modifying a direction component of a current segment of a return routing directive and adding a new segment to the return routing directive. However, the Examiner does not cite any portion of Nugent that teaches or suggests anything about a return routing directive. Instead, the Examiner refers to the rejections of claim 1, 2 and 10. Claims 1 and 10 are rejected in view of Nugent. Claim 2 does not have anything to do with a return routing directive and thus, the Examiner has cited any portion of Nugent in the rejection of claim 2 that teaches or suggests anything regarding a return routing directive. In fact, Nugent fails to mention anything at all about a return routing directive. Thus, the Examiner's reliance upon Nugent is incorrect.

Additionally, Annapareddy, Walker and Nugent teach three different routing techniques. Annapareddy teaches the use of including absolute node and group addresses in a message header so that each node in the network can use its own local routing tables to determine which I/O channel on which to send the message (Annapareddy, column 6, lines 9-40). Walker teaches a routelet based routing directive, where each routelet defines an absolute path at each node (Walker, column 7, lines 15-31). Nugent teaches a

routing directive that includes a single segment specifying the relative distance in each of the X, Y and Z dimensions that a message has to travel (Nugent, column 8, lines 28-45). The Examiner has not provided any motivation to combine the teachings of Annapareddy, Walker and Nugent. As described above regarding the rejection of claim 2, the Examiner has failed to provide a proper motivation for combining Annapareddy and Nugent. Additionally, as described above regarding the rejection of claim 10, it would not make sense to apply the routelet-based return route encoding of Walker to the routing mechanism of Annapareddy. Furthermore, the Examiner has not even attempted to provide a motivation to combine the teachings of Annapareddy, Walker and Nugent. Thus, the Examiner has failed to provide a proper rejection of claim 14.

For at least the reasons above, the rejection of claim 14 is not supported by the cited art and removal thereof is respectfully requested.

Claim 15:

Regarding claim 15, Annapareddy in view of Walker in further view of Nugent fails to teach or suggest wherein the return routing directive includes a pointer to the current segment, wherein adding a new segment to the return routing directive includes moving the pointer to the new segment. The Examiner relies upon Annapareddy and Walker and refers to the rejection of claim 4. In the rejection of claim 4, the Examiner asserts that a routing directive including a pointer to the current segment and that moving the pointer to the next segment as a part of removing the current segment are inherent in Annapareddy's system. However, as noted above regarding the rejection of claim 4, Appellants strongly disagree with the Examiner's assertion. Firstly, the Examiner has filed to provide any evidence or interpretation as to why such features would be inherent in Annapareddy. Secondly, Annapareddy's system does not include segmented routing directives encoded in a message and also does not include a pointer to the current segment as part of a routing directive. As shown above regarding claim 1, Annapareddy teaches the use of routing tables located at each node to route messages. The messages in Annapareddy do not include a routing directive including segments. Instead, a message

in Annapareddy only includes the destination node and group node addresses that are used as indexes into the routing tables in each node (Annapareddy, column 5, line 60-column 6, line 27).

Additionally, Annapareddy does not teach or suggest adding a current segment of the routing directive. In fact, Annapareddy does not teach or suggest adding any routing information from a message header during routing. Annapareddy includes the destination node and group address in messages and it would not make sense to add any new information to Annapareddy's routing information.

It is clearly not inherent in Annapareddy's system to include a routing directive that includes a pointer to a current segment of the routing directive. Nor is it inherent in Annapareddy's system to move the pointer to a new segment of the routing directive as part of adding a new segment.

Since, neither Walker nor Nugent teach or suggest anything regarding a return routing directive including a pointer to the current segment or about moving the pointer to a new segment, neither Walker, Nugent or any combination of Walker and Nugent overcome the above-noted deficiencies of Annapareddy regarding claim 15. Similarly, the Examiner's combination of Annapareddy, Walker and Nugent clearly fails to teach or suggest wherein <u>the return routing directive includes a pointer to the current segment</u>, wherein adding a new segment to the return routing directive includes <u>moving the pointer to the new segment</u>.

For at least the reasons above, the rejection of claim 15 is not supported by the cited art and removal thereof is respectfully requested.

<u>Seventh Ground of Rejection</u>:

Claims 63, 65 and 66 stand finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Flaig in view of Walker and Nugent. Appellant respectfully traverses

this rejection for at least the following reasons.

Claims 63 and 66:

Regarding claim 63, Flaig in view of Walker in further view of Nugent fails to teach or suggest decrementing the distance component for a current segment of the routing directive, wherein said incrementally encoding comprises incrementing the distance component for a current segment of the return routing directive and wherein, if after said decrementing the distance component for the current segment of the routing directive is zero, the method further comprises modifying the direction component of a current segment of the return routing directive and adding a new segment to the return routing directive so that the new segment becomes the current segment of the return routing directive when the message is sent on the selected output port, in contrast to the Examiner's assertion.

The Examiner admits that Flaig and Walker fail to teach or suggest incrementally encoding a return routing directive as recited in claim 63 and relies upon Nugent. The Examiner refers to the citing of Nugent in the rejections of claim 2 and 3. However, claims 2 and 3 recite very different limitations than claim 63. The portions of Nugent cited by the Examiner in the rejections of claims 2 and 3, namely FIG. 8 and column 14 line 1- column 15, line 14, refer only to decrementing portions of the main routing directive, but do not mention anything about *incrementing* a distance component for a current segment of a *return routing directive.* Nor does the cited portion of Nugent mention anything about adding a new segment to a *return routing directive* so that the new segment becomes the current segment of the return routing directive. Instead, the cited portion of Nugent describes only how Nugent's system uses a forward routing directive (e.g. describing a route between a source and a destination).

Thus, the Examiner has failed to cite any portion of any cited art that describes incrementally encoding a return routing directive that includes incrementing the distance component for a current segment of the return routing directive and wherein, if after said

decrementing the distance component for the current segment of the routing directive is zero, the method further comprises <u>modifying the direction component</u> of a current segment <u>of the return routing directive</u> and <u>adding a new segment to the return routing directive</u> so that the <u>new segment becomes the current segment of the return routing directive</u> when the message is sent on the selected output port. Furthermore, since Flaig and Walker, whether considered singly or in combination, fail to overcome the above noted deficiencies of Nugent, the Examiner's combination of Flaig, Walker and Nugent fails to teach or suggest the limitations of claim 63.

Additionally, regarding claim 63, the Examiner contends that it would have been obvious to a person of ordinary skill in the art to employ the teachings of Nugent within the system of Annapareddy. However, Annapareddy is not relied upon by the Examiner in the rejection of claim 63. Appellant assumes the Examiner intended to provide a motivation to combine the teachings of Nugent with those of Flaig and Walker. Regardless, the Examiner's stated motivation, namely "because by decrementing the directional component to zero allows directional limits to be set thereby triggering a change in directions such as from x-direction to Y or Z-direction" merely describes features already in the system of Flaig. No one looking to provide directional limits by decrementing a directional component to zero would not have to modify Flaig, as Flaig already teaches such functionality, and thus would not be motivated to incorporate the teachings of Nugent. Thus, the Examiner has failed to provide a proper motivation to combine the teachings of Nugent with those of Flaig and Walker.

In response to Appellant's previous arguments, the Examiner states in the Advisory Action that one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. However, in the previous response, Appellant s specifically stated, "Flaig in view of Walker does not teach or suggest" the limitations of claim 63 (the Examiner did not previously rely upon Nugent in the rejection of claim 63). Furthermore, to the extent that Appellant referred to individual references, it was to show that the Examiner's reliance on those individual references to teach a specific limitation of the claim in question was erroneous.

Therefore, for at least the reasons presented above, the rejection of claim 63 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks as those discussed above with regard to claim 63 apply also to claim 66.

Claim 65:

Regarding claim 65, Flaig in view of Walker in further view of Nugent, fails to teach or suggest wherein the return routing directive further comprises a pointer to the current segment wherein adding a new segment to the return routing directive comprises moving the pointer to the new segment. The Examiner cites Flaig, column 11, lines 12-13. However, the cited passage discusses the use cycle-stealing DMA to transfer packets between Flaig's router and memory and the use of an address pointer to read and write data in DRAM. The cited passage does not mention anything about a return routing directive, or about adding a new segment to a return routing directive including move a pointer to a new segment of the return routing directive. Neither Walker nor Nugent overcome this deficiency of Flaig.

In the Advisory Action, the Examiner responds by asserting that "Flaig teaches of pointers" and "Walker teaches the return routing directive or adding a new segment to a return routing directive" and concludes, "[t]he combinational teachings clearly suggest the claim limitation." However, as noted above, the pointers of Flaig referred by the Examiner are address pointers used to read and write data in DRAM and have nothing to do with a return routing directive. A single reference to a specific type of DRAM pointer does not in any way suggest the very different use of pointers for identifying a current segment of a return routing directive. Thus, as described above, neither Walker nor Nugent overcome this deficiency of Flaig regarding a pointer to the current segment of the return routing directive. Therefore, none of the prior art references, either singly or in combination, teach or suggest wherein the return routing directive further comprises a pointer to the current segment wherein adding a new segment to the return routing directive comprises moving the pointer to the new segment.

Thus, the rejection of claim 65 is not supported by the cited art and removal thereof is respectfully requested.

Eighth Ground of Rejection:

Claim 71 stands finally rejected under 35 U.S.C. § 103(a) as being unpatentable over Flaig in view of Brantley, Jr. et al. (U.S. Patent 4,980,822) (hereinafter "Brantley"). Appellant respectfully traverses this rejection for at least the following reasons.

Claim 71:

In regard to claim 71, Flaig in view of Brantley does not teach or suggest a storage system comprising a plurality of nodes wherein different ones of said plurality of nodes perform different functions in the storage system, wherein each one of a first portion of said plurality of nodes are storage nodes each comprising at least one mass storage device, and wherein each one of a second portion of said plurality of nodes is a host interface node configured to provide an interface for the storage system to a host computer. The Examiner admits that these limitations are not taught by Flaig. The Examiner relies on Brantley. However, Brantley describes a multiprocessing system in which all of the nodes are identical, as shown in FIGs. 1 & 2. See also, Brantley – col. 4, lines 30-59. Thus, Brantley clearly does not suggest a storage system comprising a plurality of nodes wherein different ones of said plurality of nodes perform different functions in the storage system. Moreover, routing systems such as described in Flaig and Brantley have generally been used in systems which route communication among a plurality of identical nodes, such as the homogenous multiprocessing nodes of the systems in Flaig and Brantley. The prior art does not suggest using these types of networks in heterogeneous systems where different ones of the nodes perform different functions.

In the Advisory Action, the Examiner contends that "different ones of said plurality of node perform different functions in the storage system," is subjective and not

a limitation "particularly pointing out" or "distinctly claiming" the invention. The Examiner states, "clearly a node inherently performs a function. What functions the node performs without specifically claiming the particular function is not a limitation within the claim language and therefore subjective." The Examiner is incorrect. Claim 71 clearly and precisely recites a storage system comprising a plurality of nodes, wherein different ones of the plurality of nodes perform different functions in the storage system. Claim 71 further recites that each one a first portion of the plurality of the nodes is a storage node comprising at least one mass storage device and that each one of a second portion of the plurality of nodes is a host interface node configured to provide an interface for the storage system to a host computer. Thus, rather than simply reciting that a node performs a function, as suggested by the Examiner, claim 71, particularly describes the configuration of a storage system that includes a plurality of nodes and further describes a particular relationship among those nodes.

Moreover, the Examiner's opinion that the limitation of claim 71 regarding whether different ones of the plurality of nodes performing different functions is subjective is not a proper basis for rejection. The Examiner must provide prior art that teaches the claim limitation and a proper motivation to combine the references. As noted above, routing systems such as described in Flaig and Brantley have generally been used in systems which route communication among a plurality of <u>identical nodes</u>, such as the homogenous multiprocessing nodes of the systems in Flaig and Brantley. Thus, the cited art is <u>directly in contrast</u> with claim 71 which recites that "different ones of the plurality of nodes perform different functions in the storage system" including "mass storage" and "host interface." The Examiner has failed to cite any prior art references that, whether considered singly or in combination, teach or suggest a storage system including a plurality of nodes, wherein different ones of the plurality of nodes perform different functions in the storage system. Instead, the Examiner's combination of Flaig and Brantley results in a system that includes identical nodes, each of which perform the same functions as every other node.

Furthermore, the cited art does not teach or suggest where each one of a first portion of the plurality of nodes are storage nodes each comprising at least one <u>mass storage device</u>. The Examiner cites the abstract and FIG. 2 of Brantley referring to Brantley's "associated memory module" and main store 30. However, the "associate memory module" and the main store of each node in Brantley is the processor memory, not a mass storage device. Brantley repeatedly refers to the main store 30 as a memory module. A memory module is clearly very different from a mass storage device, as mass storage devices are understood in the art. Nowhere does Brantley describe main store 30 as a mass storage device.

As Flaig does not overcome any of the above-mentioned deficiencies of Brantley, the Examiner's combination of Flaig and Brantley fails to teach or suggest a storage system comprising a plurality of nodes wherein different ones of said plurality of nodes perform different functions in the storage system, wherein each one of a first portion of said plurality of nodes are storage nodes each comprising at least one mass storage device, and wherein each one of a second portion of said plurality of nodes is a host interface node configured to provide an interface for the storage system to a host computer.

For at least the reasons above the rejection of claim 71 is not supported by the cited art and removal thereof is respectfully requested.

VIII. <u>CONCLUSION</u>

For the foregoing reasons, it is submitted that the Examiner's rejection of claims 1-6, 9-22, 24-29, 32-42, 44-63, 65, 66, 68, 70 and 71 was erroneous, and reversal of his decision is respectfully requested.

The Commissioner is authorized to charge the appeal brief fee of $500.00 and any other fees that may be due to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit

Account No. 501505/5181-68300/RCK. This Appeal Brief is submitted with a return receipt postcard.

Respectfully submitted,

Robert C. Kowert
Reg. No. 39,255
Attorney for Appellants

Meyertons, Hood, Kivlin, Kowert & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
(512) 853-8850

Date: January 30, 2006

IX. CLAIMS APPENDIX

The claims on appeal are as follows.

1. A method of sending messages in an interconnection fabric, wherein the interconnection fabric couples together a plurality of nodes, wherein each node of the plurality of nodes comprises a plurality of input ports and a plurality of output ports, comprising:

for each of a plurality of messages:

dynamically selecting a route in the interconnection fabric from among a plurality of independent routes for sending the message from a sending node to a destination node, wherein said dynamically selecting a route comprises identifying a routing directive for the selected one of the plurality of independent routes from the sending node to the destination node;

wherein said dynamically selecting a route comprises selecting different ones of the independent routes from the sending node to the destination node for at least two of the messages;

encoding the routing directive in the message, wherein the routing directive describes the route and comprises at least one segment, wherein each segment comprises a direction component and a distance component;

sending the message on one of the output ports of the sending node;

receiving the message on one of the input ports of a first node connected to the output port of the sending node;

decrementing the distance component for a current segment of the routing directive;

selecting one of the output ports of the first node according to the current segment of the routing directive in the message; and

sending the message on the selected one of the output ports of the first node.

2. The method as recited in claim 1, wherein said selecting one of the output ports comprises:

if, after said decrementing, the distance component for the current segment is greater than zero, selecting the output port corresponding to a same routing direction in which the message was traveling when received; and

if, after said decrementing, the distance component for the current segment is zero, selecting the output port corresponding to the direction component of the current segment.

3. The method as recited in claim 2, wherein if, after said decrementing, the distance component for the current segment is zero, and the output port is selected according to the direction component of the current segment, the method further comprises removing the current segment from the routing directive so that a next segment becomes the current segment when the message is sent on the selected output port.

4. The method as recited in claim 3, wherein the routing directive further comprises a pointer to the current segment, and wherein said removing the current segment comprises moving the pointer to the next segment.

5. The method as recited in claim 1, further comprising:

a subsequent node receiving the message;

the subsequent node decrementing the distance component for the current segment of the routing directive;

wherein after said decrementing:

if the distance component for the current segment is greater than zero, the subsequent node selecting the output port corresponding to a same routing direction in which the message was traveling when received; and

if the distance component for the current segment is zero, the subsequent node selecting a port corresponding to the direction component of the current segment.

6. The method as recited in claim 5, wherein the subsequent node selecting a port corresponding to the direction component comprises:

selecting the corresponding output port if the direction component for the current segment specifies a routing direction; and

selecting a device port if the direction component for the current segment specifies that the subsequent node is the destination for the message.

9. The method as recited in claim 1 wherein the interconnection fabric is a torus interconnection fabric.

10. The method as recited in claim 1, further comprising:

identifying a return route from the destination node to the sending node; and

encoding a return routing directive in the message, wherein the return routing directive describes the return route and comprises at least one segment, wherein each segment comprises a direction component and a distance component.

11. The method as recited in claim 10, further comprising calculating the return routing directive.

12. The method as recited in claim 11, wherein the interconnection fabric is bi-directional, and wherein calculating the return routing directive comprises reversing the routing directive.

13. The method as recited in claim 1, further comprising incrementally encoding a return routing directive in the message, wherein the return routing directive describes a return route from the destination node to the sending node and comprises at least one segment, and wherein each segment comprises a direction component and a distance component.

14. The method as recited in claim 13, wherein incrementally encoding comprises:

incrementing the distance component for a current segment of the return routing directive;

wherein if, after said decrementing, the distance component for the current segment of the routing directive is zero, the method further comprises modifying the direction component of a current segment of the return routing directive and adding a new segment to the return routing directive

so that the new segment becomes the current segment of the return routing directive when the message is sent on the selected output port.

15. The method as recited in claim 14, wherein the return routing directive further comprises a pointer to the current segment, wherein adding a new segment to the return routing directive further comprises moving the pointer to the new segment.

16. The method as recited in claim 1 wherein a first number of segments of a first routing directive differs from a second number of segments of a second routing directive.

17. The method as recited in claim 3 further comprising a subsequent node receiving the message and, if all of the segments of the routing directive have been removed, the subsequent node identifying itself as the destination node and selecting a device port.

18. The method as recited in claim 1, wherein each direction component comprises a direction relative to a routing direction the message was traveling in when received.

19. The method as recited in claim 1, wherein each direction component comprises an identifier of one of the output ports of one of the nodes.

20. The method as recited in claim 1, wherein the destination node is configured to communicate with a storage device.

21. The method as recited in claim 20, wherein the storage device comprises a disk drive.

22. A node, comprising:

a routing unit;

a plurality of input ports; and

a plurality of output ports;

wherein the node is configured to be connected to an interconnection fabric, wherein the interconnection fabric is configured to connect the node to a plurality of nodes;

wherein the routing unit is configured to receive a message being sent along a route from a sending node to a destination node in the interconnection fabric;

wherein the routing unit is further configured to receive a routing directive encoded in the message, wherein the routing directive describes the route and comprises at least one segment, and wherein a segment comprises a direction component and a distance component;

wherein the node is configured to receive the message on one of the input ports when the node is not the sending node, wherein the node is further configured to decrement the distance component of a current segment of the routing directive and to select one of the output ports according to the current segment;

wherein, when the node is the sending node, the node is further configured to dynamically select a route from among a plurality of independent routes from the sending node to the destination node, and wherein the node is configured to encode the routing directive for the dynamically selected

route in a message, and wherein the node is configured to send the message on one of the output ports;

wherein for at least two messages, the node is further configured to dynamically select different ones of the independent routes from the sending node to the destination node when the node is the sending node.

24. The node as recited in claim 22, wherein the node is configured to communicate with a device on a device port, wherein the device is configured to select a route, encode a routing directive in the message and communicate a message to the node on the device port when the node is the sending node.

25. The node as recited in claim 24, wherein the node is further configured to select one of the output ports according to the current segment.

26. The node as recited in claim 22, wherein the node is configured to select:

one of the output ports corresponding to a same routing direction in which the message was traveling when received if, after said decrementing, the distance component for the current segment is greater than zero; and

one of the output ports corresponding to the direction component of the current segment if, after said decrementing, the distance component for the current segment is zero.

27. The node as recited in claim 26, wherein the node is further configured to remove the current segment from the routing directive if, after said decrementing, the distance component for the current segment is zero, and the wherein the node is configured to select the output port according to the direction component of the current segment, so that a next segment becomes the current segment when the message is sent on the selected output port.

28. The node as recited in claim 27, wherein the routing directive further comprises a pointer to the current segment, and wherein said being configured to remove the current segment comprises being configured to move the pointer to the next segment.

29. The node as recited in claim 22, wherein the node is configure to select:

one of the output ports corresponding to a same routing direction in which the message was traveling when received if, after said decrementing, the distance component for the current segment is greater than zero;

one of the output ports corresponding to the direction component of the current segment if, after said decrementing, the distance component for the current segment is zero, and if the direction component for the current segment does not identify that the node is the destination node; and

a device port if, after said decrementing, the distance component for the current segment is zero and if the direction component for the current segment identifies that the node is the destination node.

32. The node as recited in claim 22, wherein the interconnection fabric comprises a torus interconnection fabric.

33. The node as recited in claim 22, wherein, if the node is the sending node, the routing unit is further configured to identify a return route from the destination node to the sending node and to encode a return routing directive in the message, wherein the return routing directive describes the return route and comprises at least one segment, wherein each segment comprises a direction component and a distance component.

34. The node as recited in claim 33, wherein, if the node is the sending node, the routing unit is further configured to calculate the return routing directive.

35. The node as recited in claim 34, wherein the interconnection fabric is bi-directional, and wherein calculating the return routing directive comprises reversing the routing directive.

36. The node as recited in claim 22, wherein the node is configured to communicate with a RAID controller.

37. The node as recited in claim 22, wherein the node is configured to communicate with a mass storage device.

38. The node as recited in claim 37, wherein the mass storage device is a disk drive.

39. A device, comprising:

an interface configured to communicate with a source node in an interconnection fabric, wherein the interconnection fabric comprises a plurality of routes between the source node and a destination node; and

a controller configured to provide a first routing directive describing a first route from the source node to the destination node, wherein the routing directive comprises at least one segment, wherein each segment comprises a distance component and a direction component, wherein the distance component is configured to be decremented by a receiving node;

wherein the controller is further configured to encode the first routing directive in a message, and to communicate the message to the source node to be sent on the interconnection fabric to the destination node; and

wherein the controller is further configured to maintain a routing table comprising a plurality of independent routes from the source node to the destination node, and wherein the controller is further configured to dynamically select the first routing directive from the routing table when communicating the message to the source node to be sent on the interconnection fabric to the destination node.

40. The device of claim 39, wherein said controller comprises a RAID controller.

41. The device of claim 39, wherein the controller comprises a host interface configured to communicate with a host computer.

42. The device of claim 39, wherein the controller comprises a disk storage device controller.

44. The device of claim 39, wherein the routing table further comprises a second routing directive describing a second route from the source node to the destination node.

45. The device of claim 44, wherein the second routing directive comprises a different number of segments than the first routing directive.

46. The device of claim 39, wherein the controller is further configured to calculate the first routing directive.

47. The device of claim 39, wherein the controller is further configured to provide a return routing directive describing a return route from the destination node to the source node, and wherein the controller is further configured to encode the return routing directive in the message.

48. The device of claim 47, wherein the controller is further configured to select the return routing directive from a routing table.

49. The device of claim 47, wherein the controller is further configured to calculate the return routing directive from the first routing directive.

50. The device of claim 39, wherein the controller is further configured to encode a return routing directive describing a return route from the destination node to the source node in the message, and wherein the return routing directive is configured to be incrementally added to as the message is routed to the destination node.

51. The device of claim 50, wherein the return routing directive is further configured to be used to return an error message to the source node if a routing error is encountered.

52. The device of claim 51, wherein the controller is further configured to use the incrementally created return routing directive to locate the routing error if an error message is returned, wherein the incrementally created return routing directive indicates a last node that successfully received the message.

53. A method of sending a message in an interconnection fabric, wherein the interconnection fabric couples together a plurality of nodes, wherein each node of the plurality of nodes comprises a plurality of input ports and a plurality of output ports, comprising:

identifying a route in the interconnection fabric for sending the message from a sending node to a destination node;

encoding a routing directive in the message, wherein the routing directive describes the route and comprises at least one segment, wherein each segment comprises a direction component and a distance component;

identifying a return route from the destination node to the sending node;

encoding a return routing directive in the message, wherein the return routing directive describes the return route and comprises at least one segment, wherein each segment comprises a direction component and a distance component;

sending the message on one of the output ports of the sending node, wherein the message includes both the routing directive and the return routing directive when sent from the initial sending node;

receiving the message on one of the input ports of a first node connected to the output port of the sending node;

decrementing the distance component for a current segment of the routing directive;

selecting one of the output ports of the first node according to the current segment of the routing directive in the message; and

sending the message on the selected one of the output ports of the first node.

54. The method as recited in claim 53, further comprising calculating the return routing directive.

55. The method as recited in claim 54, wherein the interconnection fabric is bi-directional, and wherein calculating the return routing directive comprises reversing

the routing directive.

56. A node, comprising:

a routing unit;

a plurality of input ports; and

a plurality of output ports;

wherein the node is configured to be connected to an interconnection fabric, wherein the interconnection fabric is configured to connect the node to a plurality of nodes;

wherein the routing unit is configured to receive a message being sent along a route from a sending node to a destination node in the interconnection fabric;

wherein the routing unit is further configured to receive a routing directive encoded in the message, wherein the routing directive describes the route and comprises at least one segment, and wherein a segment comprises a direction component and a distance component;

wherein the node is configured to receive the message on one of the input ports when the node is not the sending node, wherein the node is further configured to decrement the distance component of a current segment of the routing directive and to select one of the output ports according to the current segment; and

wherein, when the node is the sending node, the routing unit is further configured to identify a return route from the destination node to the sending node and to encode a return routing directive in the message, wherein the return routing directive describes the return route and comprises at least one segment, wherein each segment comprises a direction component and a distance component, wherein the message includes both the routing directive and the return routing directive when sent from the initial sending node.

57. The node as recited in claim 56, wherein, when the node is the sending node, the routing unit is further configured to calculate the return routing directive.

58. The node as recited in claim 57, wherein the interconnection fabric is bi-directional, and wherein calculating the return routing directive comprises reversing the routing directive.

59. A device, comprising:

an interface configured to communicate with a source node in an interconnection fabric, wherein the interconnection fabric comprises a plurality of routes between the source node and a destination node; and

a controller configured to provide a first routing directive describing a first route from the source node to the destination node, wherein the routing directive comprises at least one segment, wherein each segment comprises a distance component and a direction component, wherein the distance component is configured to be decremented by a receiving node;

wherein the controller is further configured to encode the first routing directive in a message, and to communicate the message to the source node to be sent on the interconnection fabric to the destination node; and

wherein the controller is further configured to provide a return routing directive describing a return route from the destination node to the source node, wherein the return routing directive comprises at least one segment, wherein each segment comprises a direction component and a distance component; and

wherein the controller is further configured to encode the return routing directive in the message, wherein the message includes both the routing directive and the return routing directive when sent from the initial sending node.

60. The device of claim 59, wherein the controller is further configured to select the return routing directive from a routing table.

61. The device of claim 59, wherein the controller is further configured to calculate the return routing directive from the first routing directive.

62. The device of claim 59, wherein the return routing directive is further configured to be used to return an error message to the source node if a routing error is encountered.

63. A method of sending a message in an interconnection fabric, wherein the interconnection fabric couples together a plurality of nodes, wherein each node of the plurality of nodes comprises a plurality of input ports and a plurality of output ports, comprising:

identifying a route in the interconnection fabric for sending the message from a sending node to a destination node;

encoding a routing directive in the message, wherein the routing directive describes the route and comprises at least one segment, wherein each segment comprises a direction component and a distance component;

sending the message on one of the output ports of the sending node;

receiving the message on one of the input ports of a first node connected to the output port of the sending node;

decrementing the distance component for a current segment of the routing directive;

selecting one of the output ports of the first node according to the current segment of the routing directive in the message;

sending the message on the selected one of the output ports of the first node; and

incrementally encoding a return routing directive in the message, wherein the return routing directive describes a return route from the destination node to the sending node and comprises at least one segment, and wherein each segment comprises a direction component and a distance component;

wherein said incrementally encoding comprises:

incrementing the distance component for a current segment of the return routing directive;

wherein if, after said decrementing, the distance component for the current segment of the routing directive is zero, the method further comprises modifying the direction component of a current segment of the return routing directive and adding a new segment to the return routing directive so that the new segment becomes the current segment of the return routing directive when the message is sent on the selected output port.

65. The method as recited in claim 63, wherein the return routing directive further comprises a pointer to the current segment, wherein adding a new segment to the return routing directive further comprises moving the pointer to the new segment.

66. A node, comprising:

a routing unit;

a plurality of input ports; and

a plurality of output ports;

wherein the node is configured to be connected to an interconnection fabric, wherein the interconnection fabric is configured to connect the node to a plurality of nodes;

wherein the routing unit is configured to receive a message being sent along a route from a sending node to a destination node in the interconnection fabric;

wherein the routing unit is further configured to receive a routing directive encoded in the message, wherein the routing directive describes the route

and comprises at least one segment, and wherein a segment comprises a direction component and a distance component;

wherein the node is configured to receive the message on one of the input ports when the node is not the sending node, wherein the node is further configured to decrement the distance component of a current segment of the routing directive and to select one of the output ports according to the current segment; and

wherein the routing unit is further configured to incrementally encode a return routing directive in the message, wherein the return routing directive describes a return route from the destination node to the sending node and comprises at least one segment, and wherein each segment comprises a direction component and a distance component, wherein in incrementally encoding a return routing directive, the routing unit is further configured to:

increment the distance component for a current segment of the return routing directive;

wherein if, after said decrementing, the distance component for the current segment of the routing directive is zero, the routing unit is further configured modify the direction component of a current segment of the return routing directive and add a new segment to the return routing directive so that the new segment becomes the current segment of the return routing directive when the message is sent on the selected output port.

68. A device, comprising:

an interface configured to communicate with a source node in an interconnection fabric, wherein the interconnection fabric comprises a plurality of routes between the source node and a destination node; and

a controller configured to provide a first routing directive describing a first route from the source node to the destination node, wherein the routing directive comprises at least one segment, wherein each segment comprises a distance component and a direction component, wherein the distance component is configured to be decremented by a receiving node;

wherein the controller is further configured to encode the first routing directive in a message, and to communicate the message to the source node to be sent on the interconnection fabric to the destination node; and

wherein the controller is further configured to incrementally encode a return routing directive describing a return route from the destination node to the source node in the message, wherein the return routing directive describes a return route from the destination node to the sending node and comprises at least one segment, and wherein each segment comprises a direction component and a distance component, and wherein the return routing directive is configured to be incrementally added to as the message is routed to the destination node, wherein the return routing directive is further configured to be used to return an error message to the source node if a routing error is encountered.

70. The device of claim 68, wherein the controller is further configured to use the incrementally created return routing directive to locate the routing error if an error message is returned, wherein the incrementally created return routing directive indicates a last node that successfully received the message.

71. A storage system, comprising a plurality of nodes interconnected by an interconnection fabric;

wherein different ones of said plurality of nodes perform different functions in the storage system;

wherein each one of a first portion of said plurality of nodes is a storage node comprising at least one mass storage device;

wherein each one of a second portion of said plurality of nodes is a host interface node configured to provide an interface for the storage system to a host computer;

wherein each node of the plurality of nodes comprises:

a routing unit;

a plurality of input ports; and

a plurality of output ports;

wherein the routing unit of each node is configured to receive a message being sent along a route from a sending node to a destination node in the interconnection fabric;

wherein the routing unit of each node is further configured to receive a routing directive encoded in the message, wherein the routing directive describes the route and comprises at least one segment, and wherein a segment comprises a direction component and a distance component; and

wherein each node is configured to receive the message on one of the input ports when the node is not the sending node, wherein the node is further configured to decrement the distance component of a current segment of the routing directive and to select one of the output ports according to the current segment.

X. EVIDENCE APPENDIX

No evidence submitted under 37 CFR §§ 1.130, 1.131 or 1.132 or otherwise entered by the Examiner is relied upon in this appeal.

XI. RELATED PROCEEDINGS APPENDIX

There are no related proceedings.